Speaker recognition performance under ideal-knowledge noise suppression: An investigation

Nilesh Madhu, Sung Kyo Jung
NXP Software, Leuven, Belgium
{firstname}.{lastname}@nxp.com

Abstract

Speaker recognition in mobile devices suffers from poor performance in noisy environments, necessitating the use of noise-suppression methods. These methods typically use time-frequency masks—optimised on the signal statistics—so as to suppress the noise components while preserving the speech. Studies in the field of speech recognition demonstrate that ideal time-frequency masks (i.e. masks generated based on ideal knowledge of the speech and noise spectra) improve the recognition rate even at very poor signal-to-noise-ratios (SNRs). The effects of such masking on the performance of speaker recognition systems are studied here, to gain a better understanding of preprocessing that is beneficial for automated speaker recognition. Two masking approaches are considered: the ideal binary mask and the ideal Wiener filter. We demonstrate that such ideal noise suppression significantly improves the recognition rate over the unprocessed system. As any noise suppression algorithm involves a trade-off between noise modulation and speech attenuation artefacts, the relative effect of these artefacts on speaker recognition performance is analysed next. We show that speech attenuation has a larger influence on the performance as compared to noise modulation at typical SNR values. Thus, we conclude, preserving speech even at the cost of lower noise suppression (and, consequently, larger noise modulation) is beneficial to speaker recognition. This conclusion is further validated.

Index Terms: Speaker recognition, ideal binary mask, ideal Wiener filter, noise suppression

1. Introduction

Robust speaker recognition in mobile devices (e.g. smartphones or tablets) is a very current problem from the industry point of view, given the proliferation of mobile devices and the use of voice as a biometric (for access control, device personalisation, etc.). Speaker recognition is typically carried out in the feature domain, with the features being extracted from a suitable time-frequency representation of the signal. Depending upon the device capability, the feature extraction and speaker recognition may either be carried out on the device itself, or only feature extraction may be carried out on the device and the recognition on central servers (i.e. the cloud).

The degradation of the input speech signal by environmental noise severely impacts speaker recognition system performance. While improved robustness to noise may be obtained by several modifications on the engine level, e.g. by multi-condition training of the speaker model [1], by making the features noise-robust [2, 3], etc., such approaches require access to both the speech engine as well as the signal acquisition feature extraction stage. This might not be practical for mobile devices. To make the system robust to noise in this case, an alternative is to denoise the input speech.

Since mobile devices are used in diverse environments, spanning a wide and dynamic range of signal-to-noise ratio (SNR) conditions, some form of noise-suppression (single- or multi-channel) is always enabled at the front-end on the signal level.

We are thankful to Dr. Patrick Naylor of Imperial College, London, for his assistance and advice on this study. We are grateful to Jón Guðnason, Patrick Naylor and Mike Brookes for making their speaker identification framework and their Voicebox toolbox available online.

These systems typically use a temporal-spectral mask (also called a gain function) as the standard noise suppression technique for single-channel noise suppression, and as a postprocessor for multi-channel noise suppression. The masks are applied to an appropriate time-frequency representation of the microphone signals, such as the short-time Fourier transform (STFT) representation. Computation of the masks requires estimates of the noise and the speech power spectral densities. Our study considers the case where these power spectral densities are perfectly known (ideal-knowledge). The masks derived using such ideal-knowledge are termed as ‘ideal’ masks. In addition to providing an insight into the processing required to improve speaker recognition system performance, ideal spectral masking also yields an upper-bound on the improvement achievable using practical noise suppression algorithms. This knowledge can be translated to improving the built-in noise suppression module of mobile devices which, normally, is tuned towards improving the intelligibility and listening comfort of voice calls.

2. Signal model and mask definition

The signal model we consider is that of a target speech signal embedded in additive noise. Thus, in the discrete time domain we have:

\[ x(n) = s(n) + v(n) \]

\[ x(n) = s(n) + v(n) \]

where \( x(n) \) is the observed microphone signal, \( s(n) \) is the speech signal, \( v(n) \) is the additive noise and \( n \) is the sample index in time. For noise suppression, usually a windowed STFT representation of the signals is considered as below:

\[ X(k, l) = S(k, l) + V(k, l) \]

\[ X(k, l) = S(k, l) + V(k, l) \]

with \( k \) being the discrete frequency index and \( l \) being the index of the time-frame under consideration.

State-of-the-art noise suppression algorithms (see, e.g. [5–8]) compute a time-frequency mask \( M(k, l) \) based on estimates of the power spectral density (PSD) of the noise, \( \Psi_{SS}(k, l) \) and the speech PSD \( \Psi_{SS}(k, l) \) and apply this mask to the spectrum of the (noisy) input signal. The noise-suppressed signal is then reconstructed from this masked spectrum, using standard overlap-add or overlap-save reconstruction. Note that only the amplitude of the spectrum is subject to the mask, and is used in conjunction with the original (noisy) phase of \( X(k, l) \) for reconstructing the noise-suppressed signal.

An ideal mask is obtained if the true values of \( \Psi_{VV} \) and \( \Psi_{SS} \) are known for each time-frequency point \((k, l)\). Two well-known variants of ideal masks have been extensively studied in the field of speech recognition: the ideal binary mask (IBM) [9] and the ideal Wiener filter (IFW) [10]. Due to the significant improvements in speech intelligibility reported when using these masks, these are also the algorithms we consider in this study.

\[ 1 \text{We shall only consider the single-channel variant here, since multi-channel approaches can be reduced to this case after the application of a suitable linear first stage (e.g. a filter-and-sum beamformer (FSB), a generalised sidelobe canceller (GSC), etc. See e.g. [4] for a very accessible exposition of these and other beamforming approaches).} \]

\[ 2 \text{State-of-the-art noise suppression algorithms are usually based on the MMSE or MAP estimators of clean speech amplitude. Depending upon the assumptions made on the signal statistics, this gives rise to a} \]
2.1. Ideal binary mask (IBM)

The IBM is defined as:

$$M_{IBM}(k, l) = \begin{cases} 1 & \frac{\Psi_{SS}(k, l)}{\Psi_{VV}(k, l)} > \Upsilon \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where $\Upsilon$ is the decision threshold. When $\Upsilon = \text{SNR}(k)$, where SNR($k$) is the input signal-to-noise ratio at frequency $k$, the resulting $M_{IBM}(k, l)$ is termed as the IBM with a local criterion (IBM-LC) [11]. An important property of the IBM-LC is that, for a given noise and speech signal, the mask pattern remains the same, independent of the mixing SNR. The IBM-LC has been shown to give a remarkable improvement in speech intelligibility for very low input SNRs [10, 11].

When $\Upsilon$ is a fixed value, selected independently of the data, the resulting mask is termed as IBM with a fixed criterion (IBM-FC) [9]. The usual value selected for the fixed threshold is 1 (corresponding to a 0 dB threshold). The mask pattern of the IBM-FC differs with the input SNR.

For SNRs above 0 dB, the IBM-FC outperforms the IBM-LC in terms of speech intelligibility improvement, however, for input SNRs below 0 dB, the IBM-LC tends to outperform the IBM-FC. This has been demonstrated in several studies (see, e.g. [10, 11]). For the purpose of this study, therefore, we use a modified IBM as follows:

$$\Upsilon = \begin{cases} 1 & \text{SNR} > 0 \text{ dB} \\ \text{SNR}(k) & \text{SNR} \leq 0 \text{ dB} \end{cases} \quad (4)$$

where SNR represents the broadband (global) input SNR. In subsequent usage, it is the binary mask generated using the threshold in (4) that we refer to as the IBM. Such a mask combines the advantages of IBM-FC and IBM-LC and is best suited for our purposes.

2.2. Ideal Wiener filter (IWF)

The IWF gain function is defined as:

$$M_{IWF}(k, l) = \frac{\Psi_{SS}(k, l)}{\Psi_{SS}(k, l) + \Psi_{VV}(k, l)} \quad (5)$$

Unlike the IBM function, the IWF generates a soft mask $M_{IWF}(k, l) \in [0, 1]$. The IWF has been demonstrated, in [10], to yield almost perfect speech intelligibility improvement for different SNRs and different noise types, in subjective recognition experiments.

2.3. Power spectral density estimates

Whereas in practice the power spectral densities $\Psi_{SS}(k, l)$ and $\Psi_{VV}(k, l)$ are estimated by recursive smoothing of the periodogram, in studies of the IBM and the IWF these quantities are computed as the instantaneous squared amplitudes of the respective signals:

$$\Psi_{SS}(k, l) = |S(k, l)|^2,$$

$$\Psi_{VV}(k, l) = |V(k, l)|^2 \quad (6).$$

These are also the estimators of the power spectral densities that we use in our study.

3. Experimental setup

3.1. Database and baseline

All experiments were carried out on the YOHO corpus [12], a large scale speech corpus supporting text-dependent and text-independent speaker recognition and verification research. It family of gain functions (see e.g. [5–7]), of which the Wiener filter is a member. Under ideal-knowledge all these algorithms perform similarly, but the Wiener filter is easier to parameterise for the study. contains speech data (series of three two-digit combination-lock phrases) from 108 male and 30 female speakers. The data is divided into a training and a test subset. Training data was collected in 4 sessions with 24 phrases per session. Test data was gathered in 10 sessions with 4 phrases per session. The data was sampled at 8 kHz, at a sample depth of 12 bits.

We used data from all four training sessions to train the speaker models. The features used for speaker modelling were the classic 12-dimensional Mel-frequency cepstral coefficients (MFCC’s) $C_i$, $i = 1, 2, \ldots, 12$. The coefficient $C_i$, representing the energy of the frame, was discarded. The features were extracted from speech segments of length 32 ms, and a frame shift between adjacent segments of 10 ms. Before extracting the features, the speech data was segmented to remove leading and trailing speech pauses, such that only the segment containing the combination-lock phrase was used in the training and test phases. The statistical distribution of the features was modelled by Gaussian mixture models. The training and evaluation was done with the help of the speaker identification toolbox customised for the YOHO database, and available freely at [13].

For the baseline we first modelled the feature vectors by a 32-component Gaussian mixture model (GMM) with a diagonal covariance matrix structure. The recognition rate we obtained with this on clean speech was approximately 95%, which is close to that reported in [14] for the YOHO database. Using a 128-component GMM with, again, a diagonal-covariance matrix structure only increased performance marginally (to approximately 98%). In contrast we found that using a 16-component Gaussian mixture model with a full covariance matrix structure yielded a recognition rate of approximately 98%, so we used this setting as the baseline setting for further tests. An overview of the system performance as a function of model size and parametrisation is presented in Figure 1.

![Figure 1: Effect of number of components in the GMM on the recognition rate for clean speech data.](image)

3.2. Generation and pre-processing of the noisy signals

The performance of the speaker recognition system was tested in three different noise conditions. The first three noise types: pink, babble and Volvo (labelled here as ‘car’) were taken from the NOISEX-92 database [15, 16]. The fourth noise type was a recording made on the side of a motorway of passing traffic. This is a highly non-stationary noise condition, and consists of a rather low background (stationary) noise level superimposed with intermittent, high energy, broadband ‘swish’-ing noises as cars pass the spot where the recording microphone is located. This recording was made in-house.

The mixing SNRs ranged from -30 dB to 30 dB in steps of 5 dB, i.e. SNR $\in \{-30, -25, \ldots, 30\}$. The noise signals were synthetically mixed with the target speech signal for all the mixing SNRs. For each combination of speaker data, SNR and noise type, masks were computed based on the clean speech and noise spectra according to (5) for the IWF and according to (4) for the IBM. For completeness, the IBM-LC was tested only
for the positive SNRs (for negative SNRs the IBM is the same as the IBM-LC). These masks were then applied to the noisy mixture and the noise-suppressed signal was resynthesised from the masked spectrum. This signal was then fed into the speaker recognition framework and the results were computed for each combination of mask function, noise-type and mixing SNR.

The signal spectra were obtained by computing a windowed, 512-point STFT on frames of 512 samples length. The frame-shift between adjacent frames was 256 samples. The resynthesis was done using a synthesis window in the standard overlap-add framework. The element-wise square-root of a 64 ms long von Hann window was used for the analysis and the synthesis.

4. Experiments and analyses

4.1. Testing the effect of IBM and IWF pre-processing

Figure 2 depicts the performance of the speaker recognition system when fed with the unprocessed (noisy) signal, the IBM-enhanced signal, the IWF-enhanced signal and the IBM-LC-enhanced signal. As expected, for very low SNRs, the recognition rate of the reference system (unprocessed signal) is around 0.73%, which is little better than a random guess. The performance improves steadily with increasing SNR, but remains rather low until the SNR is around 25 dB. Clearly, such performance is unacceptable for mobile devices.

IBM-processing consistently improves the performance of the speaker recognition system at all SNRs. Even at the worst SNRs, speaker recognition with IBM processing is around 10%–20%, which is significantly better than a random guess. The performance when using IWF-based noise suppression is, in turn, significantly better than the IBM approach at all SNRs. In contrast to the IBM or the unprocessed signals, the IWF-processed signals would deliver a speaker recognition performance comparable to clean speech input already at SNR levels of around 10–15 dB. However, no approach yields an acceptable recognition rate for all the SNR conditions that would be commonly found in practice (approximately the range from 5 dB to 30 dB). Additionally, all approaches suffer from the same two artefacts of time-frequency masking: modulation of the output signals by noise (coming from time-frequency regions with a high M(k, l) values) and speech attenuation (coming from speech regions with low M(k, l) values).

Note that for SNR ≤ 0 the IBM reduces to IBM-LC, with the mask pattern being the same for a given combination of noise and speech signals. The reduction in the recognition rate with decreasing SNR may be the result, therefore, of the modulation of the masked speech by the noise at the time-frequency points where M_{IBM}(k, l) = 1. In contrast, the performance with IBM-LC enhanced signals, plotted only for positive SNRs, remains rather flat, even at high SNRs. Note that the mask function remains the same for IBM-LC, irrespective of the mixing SNR. At higher SNRs then, the bad performance of the IBM-LC approach could be due to the attenuation of time-frequency regions containing speech. In contrast, the IBM approach, which for positive SNRs functions as the IBM-FC with a 0 dB threshold, would mask increasingly fewer time-frequency points as the SNR increased, implying less attenuation of time-frequency points containing speech. The recognition rate also increases sharply, indicating that decreased speech attenuation could indeed be the factor influencing performance here. The consistent superiority of the IWF over the IBM could be due to the fact that the IWF gain function is soft and has a built-in trade-off between the noise modulation and speech attenuation artefacts, which degrade the performance of the IBM.

4.2. Effect of noise modulation and speaker attenuation

As stated in Section 4.1, there are two artefacts as a consequence of time-frequency masking, which degrade the performance of noise-suppression algorithms: noise modulation and speech attenuation. Evidently, the best attainable performance of noise suppression algorithms would depend upon a trade-off between these factors. This trade-off depends upon the relative influence of each factor. This is examined here. We consider only the IBM for this analysis because the binary nature of the decisions make it easier to analyse the relative effect of each type of artefact. With respect to the IBM we have:

1. speech attenuation where M_{IBM}(k, l) = 0 on time-frequency points that contain speech, and
2. noise modulation where M_{IBM}(k, l) = 1 on the time-frequency points containing speech

Since both effects are simultaneously present when using the IBM mask, to analyse the effect of only one factor we must somehow compensate for the other. We do this as follows: to analyse only the effect of speech attenuation, we compute the mask as usual for the IBM approach ((3) using the threshold in 4). Next, for all regions where M_{IBM}(k, l) = 1, we replace the noisy spectral amplitude by the amplitude of the clean speech spectrum. The signal is then resynthesised and fed to the speaker recognition framework. Here there is no noise modulation, only speech attenuation. We call this the no noise modulation scenario. Similarly, to analyse the effect of noise modulation we replace the noisy spectral amplitude in time-frequency regions where M_{IBM}(k, l) = 0 by the clean speech spectral amplitude. The signal is then resynthesised and evaluated by the speaker recognition framework. Here only noise modulation is present. We call this the no speech attenuation case.

The results are depicted in Figure 3. The results using IWF and the IBM are also plotted for ease of comparison of the performance. It may be seen the mask with no noise modulation performs better than the IBM for all SNRs. The performance difference is slight for higher SNRs, and only becomes significant for low SNR conditions. For the extremely low conditions, the performance of this mask is even better than that of the IWF, indicating that the IWF, despite its soft/continuous trade-off, is also influenced by noise modulation. In contrast, the mask with no speech attenuation demonstrates the kind of performance we would like to achieve in a real system. Indeed at positive SNRs, the performance converges to performance in clean speech. At lower SNRs, the performance degrades, but still remains better than that of all the other approaches. We may conclude here that in moderate to high SNRs conditions, it is the speech attenuation that drives the recognition performance. The speaker recognition system seems to be quite sensitive to speech attenuation.

On the other hand, noise modulation only has a moderately deleterious effect on the performance. Only when the
The factor $\eta$ is really poor (i.e. $\text{SNR} \ll 0 \text{dB}$) does the effect of noise modulation on the recognition performance become noticeably large.

### 4.3. Effect of trading-off noise modulation and speech attenuation

Based on the analysis in Section 4.2 we hypothesise that reducing the speech attenuation – even at the cost of more noise modulation – should yield better recognition performance where speech attenuation dominates recognition performance, whereas the performance should degrade where noise modulation dominates the recognition performance. To verify this, we modified the IWF and IBM to allow for more speech preservation by the mask function at the cost of increased noise modulation. This was done by generalising (4) and (5) as follows:

$M_{\text{IBM-Relax}}(k, l) = \begin{cases} 
\frac{\Psi_{SS}(k, l)}{\Psi_{SS}(k, l) + \eta \Psi_{VV}(k, l)} & \text{SNR} > 0 \text{ dB} \\
1 & \text{SNR} \leq 0 \text{ dB} 
\end{cases}$

$M_{\text{IWF-Relax}}(k, l) = \frac{\Psi_{SS}(k, l)}{\Psi_{SS}(k, l) + \eta \Psi_{VV}(k, l)}$  (8)

The factor $\eta$ in (7) and (8) controls the amount noise suppression. The original IWF and IBM masks are obtained for $\eta = 1$. Setting $\eta < 1$ indicates less aggressive noise suppression and more speech preservation. For the test $\eta$ was set to 0.5, i.e. ‘under-estimating’ the noise floor by 6 dB. The results obtained using these relaxed mask functions are depicted in Figure 4 and are contrasted with the original IWF and IBM processing. The results validate our hypothesis. Consistent with the observations in Section 4.2, as the SNR decreases, the system becomes more sensitive to noise modulation. Thus relaxing the noise suppression leads to worse performance. This is demonstrated by the intersection of the IBM-relax and the IBM performance characteristics at lower SNRs (clearly visible for the car-noise). The Wiener function also demonstrates a similar behaviour: the IWF-relax and the IWF performance-curves show little difference at the low SNRs. For moderate to high SNRs, however, where speech attenuation is the major factor affecting system performance, relaxing the noise suppression (and thereby preserving more speech components) improves the recognition rates by as much as 10%–15%. The improvements are maximal for the SNRs in the middle of the tested range. This seems to be the region where the system is most sensitive to the trade-off between noise suppression and speech attenuation.

### 5. Conclusions

The effect of noise suppression by ideal time-frequency masks on the performance of a speaker recognition system is presented. Two time-frequency masking strategies were studied: the ideal Wiener filter (IWF) and the ideal binary mask (IBM). Speaker recognition rates improved when pre-processing the input signals by such ideal masks, as compared to using the unprocessed signal. However, neither strategy yields good results for the SNR range likely to be encountered in practice ($\approx \text{SNR} \geq -5 \text{ dB}$). This is because all time-frequency masking approaches introduce two kinds of artefacts: speech attenuation, which occurs when the mask is low at a time-frequency point that contains speech; and noise modulation, which occurs when the mask-value is high. The design and tuning of noise suppression algorithms typically involves finding the best trade-off between speech attenuation and noise suppression. This trade-off depends upon the relative importance of the artefacts to the application under consideration, and is, therefore, application specific. The relative influence of noise modulation and speech attenuation for speaker recognition was studied using the IBM. We found that for moderate to high SNRs ($-15 \text{ dB} < \text{SNR} \leq 30 \text{ dB}$), speech attenuation was the major factor limiting system performance whereas noise modulation only exercised a low influence. For the SNRs on the lower end of the spectrum, noise modulation was the major factor affecting system performance. We hypothesised that relaxing noise suppression, thereby preserving more speech, should lead to improved recognition performance at SNRs where speech attenuation dominates the recognition performance and degrade the performance where noise modulation exercises the major influence. We tested this hypothesis by appropriately modifying the IWF and IBM. The results validate our hypothesis. Furthermore, the sensitivity to speech attenuation seems most pronounced in the range of SNRs typically encountered in practice. These results provide valuable guidance for the design and tuning of noise suppression algorithms for speaker recognition applications.
6. References


